



Středoškolská technika 2017

Setkání a prezentace prací středoškolských studentů na ČVUT

Měření frekvence pomocí mikrokontroléru ATmega328

Ondřej Nevosad

**Střední škola průmyslová, technická a automobilní Jihlava
tř. Legionářů 3, 58601 Jihlava**

STŘEDNÍ ŠKOLA PRŮMYSLOVÁ, TECHNICKÁ A AUTOMOBILNÍ JIHLAVA



Měření frekvence mikroprocesorem

Jihlava 2017

Autor

Ondřej

Obsah

Úvod.....	5
1. Kmitočet a způsoby jeho měření	6
1.1 Co je to Kmitočet.....	6
1.2 Měření a výpočet kmitočtu	6
2. Metody měření frekvence (kmitočtu)	7
2.1 Analogové metody.....	7
2.1.1 Vibrační metoda.....	7
2.1.2 Záznejsová metoda	8
2.2 Digitální metody	9
2.2.1 Čítačová metoda	9
2.2.2 Metoda měření délky periody	10
2.3 Měření osciloskopem.....	11
2.3.1 Analogový osciloskop.....	11
2.3.2 Číslicový osciloskop	12
2.3.3 Měření kmitočtu osciloskopem.....	13
3. Vlastní výrobek.....	14
3.1 Arduino	15
3.2 Použité součástky.....	15
3.2.1 Arduino Nano	15
3.2.2 LCD Displej.....	17
3.2.3 Stabilizátor napětí	17
3.2.4 Ochrana vstupních pinů	18
3.2.5 Dutinkové lišty.....	18
3.2.6 Kondenzátory.....	19
3.2.7 Potenciometr a trimry	19

3.2.8	Magnetický snímač	20
3.2.9	Ochrana proti přepólování	20
3.3	Seznam součástek - tabulka	21
3.4	Popis činnosti	23
3.5	Simulace a vývoj	24
3.6	Deska plošných spojů	25
3.7	Uvedení do provozu	26
4.	Zajímavé části zdrojového kódu	27
4.1	Měření otáček	27
4.2	Měření napětí	27
4.3	Zobrazení otáček	28
4.4	Zobrazení napětí	29
4.5	Zobrazení jednotek	30
4.6	Velký font číslic	31
4.6.1	Části číslic	31
4.6.2	Stavba číslic	31
4.6.3	Přiřazení číslic k hodnotě	32
5.	Další možná řešení	33
5.1	Snímání indukčním čidlem	33
5.2	Realizace převodníkem f/U	33
6.	Závěr	33
7.	Zdroje informací	35
7.1	Schéma zapojení	36
7.2	Plošný spoj	36
7.3	Osazovací plán	37

Seznam tabulek

1	Seznam součástek	21
---	------------------------	----

Úvod

Nápad vytvořit otáčkoměr vznikl v loni v létě, když si můj kamarád stavěl malotraktor, aby se lépe dostal do hustého lesa pro dříví. Po zhotovení základní elektroinstalace mě požádal, jestli by nešlo vyrobit něco jako palubní počítač, který by mu ukazoval aktuální otáčky motoru. Nechtěl totiž malý diesel zatěžovat vysokými otáčkami a zároveň aby neměl malé otáčky při tahu břemene. Slovo dalo slovo a pustil jsem se do návrhu otáčkoměru. Zprvu jsem se bránil mikroprocesorové technice a proto jako prvotní plán měl být otáčkoměr realizován tvarovačem impulzů a zobrazování mělo být přes obvody LM3914 a řadou nebo půlkruhem z několika LED, které by se rozsvěcovaly s rostoucími otáčkami. Od tohoto plánu bylo velmi rychle odstoupeno kvůli malé spolehlivosti a složitosti vytvoření DPS. Jeden z návrhů měl být realizován pomocí převodníku frekvence/napětí sestaveného z operačních zesilovačů. Šlo by vlastně o zdroj napětí řízený frekvencí, tudíž zobrazování otáček by bylo realizováno voltmetrem. Kvůli velkým rozměrům DPS bylo řešení zavrhnuto. Další návrh padl na zobrazování 7-segmentovým LED displejem, ten se při testování projevil jako nevhodný pro použití na intenzivním světle. Nakonec i přes můj odpor padla volba na digitální technologie. Zařízení bylo realizováno mikroprocesorem Atmel v platformě Arduino a zobrazování LCD displejem určeným pro tyto platformy. Toto umožnilo jednoduché snímání otáček a možnost měřit i další parametry (v našem případě správná činnost dobíjecí soustavy).

I přes toto finální rozhodnutí v úvodní části práce princip všech výše zmiňovaných metod uvádím. Snad tento text umožní dalším studentům, kteří se budou snažit proniknout do této problematiky, lépe pochopit podstatu mé práce. Pro zkušeného čtenáře, který je obeznámen s obecnými principy elektroniky je studium těchto pasáží zbytečné a doporučuji začít práci číst až od strany 15 kapitoly „Vlastní výrobek“.

1. Kmitočet a způsoby jeho měření

1.1 Co je to Kmitočet

Kmitočet je veličina, která udává, kolikrát se určitá perioda zopakuje za daný časový úsek např.: počet úderů srdce za minutu. Jeho základní jednotka je 1 Hz (Hertz). Jednotka patří do soustavy SI a byla pojmenována pojmenovaná po Heinrichu Rudolfu Hertzovi, který se zabýval elektromagnetickým vlněním. Jiný název kmitočtu je také Frekvence. Vyskytuje se ve střídavých elektrických obvodech, kde může mít různý tvar, nejčastěji sinus, obdélník a pila.

1.2 Měření a výpočet kmitočtu

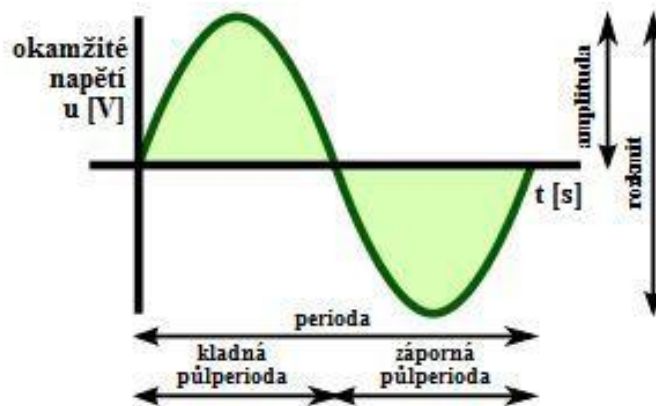
Pokud chceme měřit kmitočet, můžeme výsledku dosáhnout několika způsoby. Analýzou průběhu můžeme zjistit počet impulzů za daný časový úsek (v případě že časový úsek měření bude 10s a napočítáme 20 impulzů, výsledná frekvence tohoto průběhu bude 2Hz), nebo zjistíme dobu trvání periody průběhu a podle vztahu uvedeného níže vypočítáme.

Perioda je část průběhu, která se neustále opakuje (periodicky). Trvá takovou dobu, než se průběh dostane zpět do výchozího stavu (obr. 1.). Existuje několik metod pro měření kmitočtu. Lze je rozdělit na digitální (číslíkové) a analogové metody.

$$f = \frac{1}{T}$$

f - výsledná frekvence [Hz]

T - délka periody [s]



1 Perioda

2. Metody měření frekvence (kmitočtu)

Metody měření kmitočtu lze rozdělit podle principu zpracování údajů na Analogové a Digitální.

Pro velké množství měřících metod budou uvedeny jen některé příklady.

2.1 Analogové metody

Měřiče pracující na principu analogových metod měří přímo celý průběh signálu, který převedou na „zpracovatelnou“ veličinu (z pravidla napětí) a zobrazí příslušnou hodnotu na ocejchovaném ukazateli (obr. 2). Některé analogové měřiče jsou mechanické a nepotřebují napájecí napětí. I přes nízké provozní a pořizovací náklady jsou dnes vytlačovány digitálními měřiči.

2.1.1 Vibrační metoda

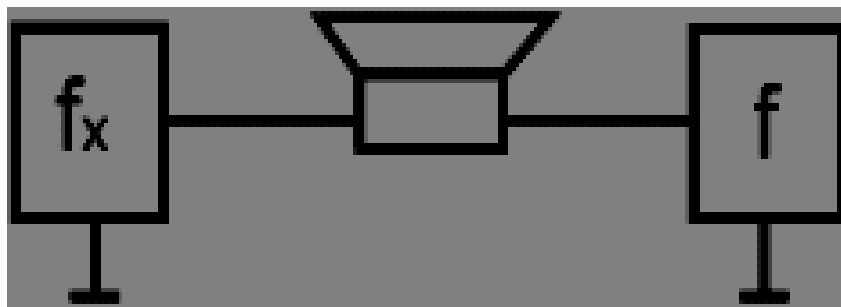


2 Ukazatel

Vibrační metoda funguje na základě kovových jazýčků, které jsou vsazeny do elektromagnetického pole cívky a naladěné na danou frekvenci. Proud se střídavým průběhem prochází cívkou, kde vytvoří elektromagnetické pole a rozkmitá jazýček, jehož mechanická frekvence odpovídá rezonanci magnetického pole. Tato metoda je náchylná na změnu teploty.

2.1.2 Zázňejová metoda

Zázňejová metoda je metoda porovnávací. K měření se využívá sluchátko (reproduktor) do kterého se z generátoru přivede známý kmitočet (f) a zároveň neznámý kmitočet (f_x) podle schématu (obr. 3) Známý kmitočet se ladí tak dlouho než uslyšíme čistý tón. Neznámý kmitočet se tedy rovná vyladěnému známému kmitočtu.



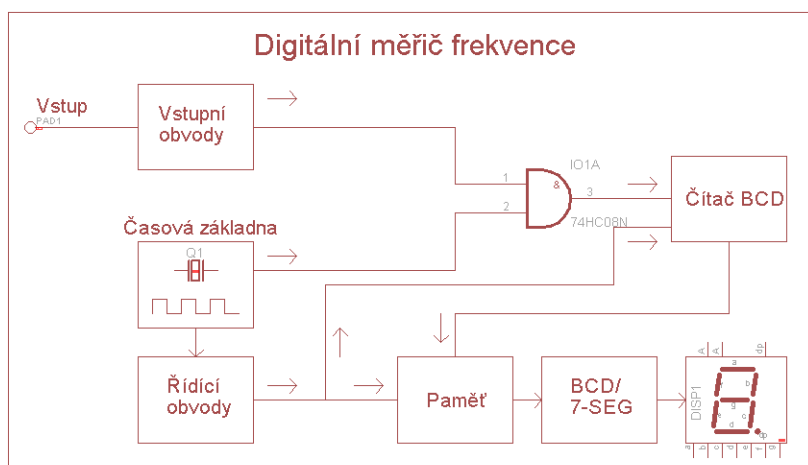
3 Zázňejová metoda

2.2 Digitální metody

Digitální (číslicové) metody jsou dnes nejrozšířenější. Pracují tak že z měřeného signálu odeberou pouze jednotlivé vzorky, které se v paměti zpracují a podle zadaných vztahů přepočítají na pravdivou hodnotu. Ta se po určitém cyklu zobrazuje na displeji např. 7-segmentová soustava displejů nebo LCD.

2.2.1 Čítačová metoda

Metoda pracuje na principu čítání impulzů, které se opakovaně děje po určitou dobu, kterou udává vnitřní oscilátor přístroje. Po uplynutí čítací doby se „uzavře“ vstup a načítané impulzy se v paměti zpracují, zobrazí na displeji, dojde k vynulování čítačů a celý cyklus se opakuje (viz obr. 4). Tato metoda se často realizuje mikroprocesorem.



4 čítačová metoda

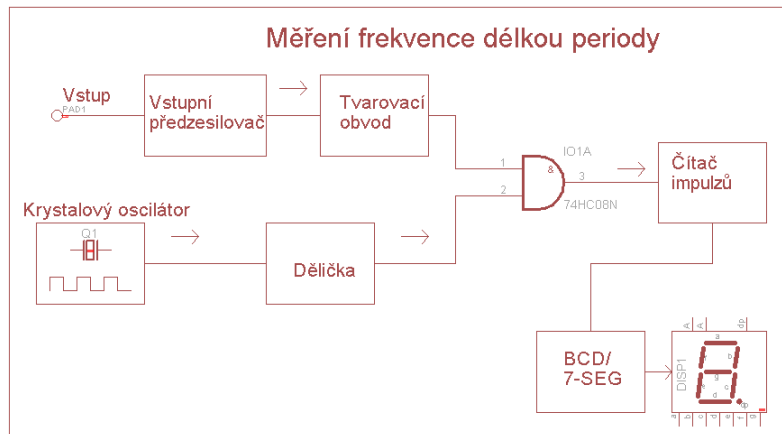
2.2.2 Metoda měření délky periody

Metoda pracuje tak že na vstup se přivede signál a tvarovací obvod převede jeho periodu na jeden pravouhlý impuls, který udává časovou základnu. Po délku trvání této periody prochází přes „AND“ hradlo impulzy z Krystalového oscilátoru. Ty se

načítají do čítače a převedou podle vzorce $f(x) = \frac{1}{T(p)}$ na frekvenci, která se zobrazí na displeji.

T_p je délka periody vstupního signálu vypočítá se z počtu načítaných impulsů (n) a známé frekvence oscilátoru f_k

$$T_p = \frac{n}{f_f}$$



5 metoda délky periody

2.3 Měření osciloskopem

Osciloskop je měřicí přístroj, který průběh signálu vykresluje na obrazovce, takže signál můžeme pozorovat okem. Podle stupnice na obrazovce odečítáme amplitudu a délku periody. Z těchto údajů pak získáváme frekvenci. Díky grafickému zobrazení lze osciloskopy měřit i fázový posun dvou signálů a charakteristiky součástek.

2.3.1 Analogový osciloskop

Základem analogového osciloskopu je časová základna pilového průběhu, která „pohybují“ svislými vychylovacími deskami a tím na obrazovce vychyluje rozsvícený paprsek ve vodorovné ose „X“. Po přivedení libovolného signálu na vstup se přes vnitřní obvody začnou „pohybovat“ i vodorovné vychylovací desky a tím se bod pohybuje i ve svislé ose „Y“. Složením průběhu časové základny a přivedeného signálu se bod bude pohybovat v reálném tvaru vstupního signálu po ocejchované obrazovce.

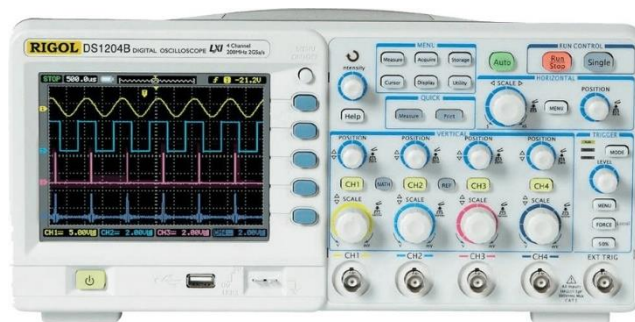


6 Analogový osciloskop

2.3.2 Číslicový osciloskop

Základem číslicového osciloskopu je A/D převodník. Ten převede analogový průběh na tzv. teplotní kód. Následně se převede na binární a tak se uloží do paměti. Zobrazování zajišťují algoritmy, které porovnávají pozici na displeji s hodnotami uloženými v paměti. Podle tohoto klíče se pak rekonstruovaný signál zobrazí na LCD displeji.

Podle Shannonova teorému platí, že vzorkovací frekvence při rozkladu signálu musí být více než 2x větší než frekvence vstupního signálu. Jinak signál nelze rekonstruovat zpět. Tzn. Pokud bude vstupní signál 45kHz, musí být vzorkovací frekvence více než 90kHz.

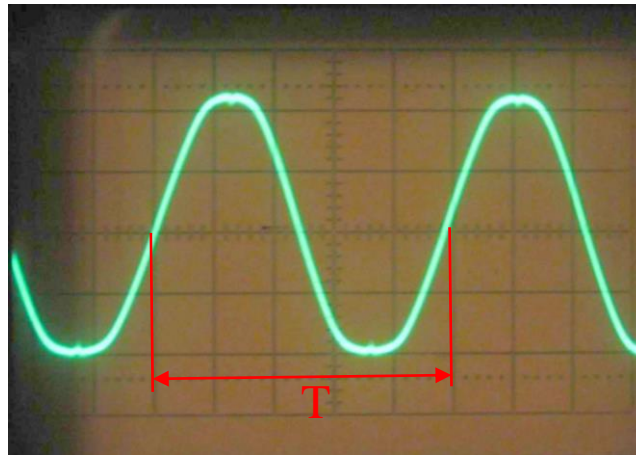


7 Číslicový osciloskop

2.3.3 Měření kmitočtu osciloskopem

Měření kmitočtu probíhá tak že se na vstup osciloskopu přivede libovolný signál. Vhodně se nastaví velikost časové základny a výška amplitudy tak, abychom viděli alespoň jednu periodu. U číslicového Číslicové osciloskopu pouze odečteme měřenou frekvenci na obrazovce. U analogových musíme spočítat počet dílků na periodu a podle nastavení časové základny spočítáme délku a dosadíme do již známého vzorce. Na obrazovce nyní vidíme, že délka periody „T“ je 5 dílků. Časová

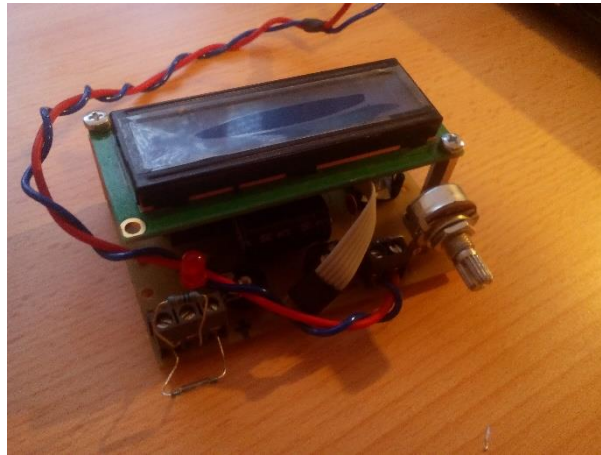
základna je nastavená na 10ms/dílek. Dosadíme do vzorce: $f = \frac{1}{T}$

$$f = \frac{1}{5 * 10 * 10^{-3}} = 20Hz$$


8 Sinus

3. Vlastní výrobek

Jako demonstrační výrobek byla zvolena výroba otáčkoměru pro traktory, starší nákladní automobily, stavební stroje a malé motocykly. Srdcem otáčkoměru je Arduino Nano s mikroprocesorem Atmel Mega 328P. K zobrazování byl použit LCD Displej 16x2 znaky na kterém se objevují otáčky v rozsahu 0-9900 ot/min a palubní napětí se signalizací přebíjení, nebo nedostatečného dobíjení. Další výborná vlastnost je regulace jasu displeje, která zamezuje oslňování řidiče v nočních hodinách. Protože frekvenční rozsah pro 9900 ot/min je 165 Hz, není nutné počítat s odchylkami při vysokých frekvencích, lze zvolit takřka libovolnou metodu měření. Proto je měření otáček realizováno metodou měření délky periody. Jako snímač byl zvolen snímač magnetického pole, který se vloží do blízkosti klikové hřídele. Na té musí být umístěn podobný magnet, jako je u cyklistického tachometru.



9 Otáčkoměr

3.1 Arduino

Arduino je malý jednočipový počítač založený na procesorech firmy Atmel konkrétně řada AT-Mega. Jde o vývojovou platformu, kterou lze použít jako samostatný mikroprocesor. Jeho hlavní výhodou je jednoduchost připojení vstupních i výstupních periférií a programování přes USB rozhraní. Je totiž vybaveno převodníkem USB/RS232 ve kterém je nahraný bootloader, který umožňuje jednoduché programování. Programování se provádí v jazyce „C“ ve vývojovém prostředí Arduino IDE, které je zdarma dostupné i v českém jazyce. Pro Arduino existuje spousta knihoven a příkazů speciálně pro tuto platformu, které velmi zjednodušují programování. Arduino se vyskytuje v mnoha variantách, některé mají již své předurčení jako například model Esplora, která je vyráběna jako hrací konzola. Pro tuto platformu se vyrábí i spousta shieldů. Není problém sehnat Ethernet shield nebo shield pro bluetooth, paměťové karty, a nebo ultrazvukové snímače vzdálenosti Ty jsou vyráběné podle rozměrů Arduina a z pravidla k nim existují i ovládací knihovny. Tudiž je stačí jen osadit lámací lištou a zasadit do hlavní desky. Arduino se může chlubit různorodostí vstupních i výstupních pinů. Digitální piny jsou vstupně/výstupní a některé z nich jsou konstruované i pro PWM regulaci. (jsou označeny „~“). K výbavě také patří komunikační (RX/TX) a analogové vstupní piny. Arduino disponuje velkou pamětí a oscilátory, ale bohužel také vysokým pořizovacími náklady.

3.2 Použité součástky

Tato kapitola je věnována popisu součástek, ze kterých byl otáčkoměr postaven. V kapitole jsou popsány všechny důležité součástky. Od samotného srdce otáčkoměru až po ochranu proti přepólování.



10 Arduino Esplora a Mega2560

3.2.1 Arduino Nano

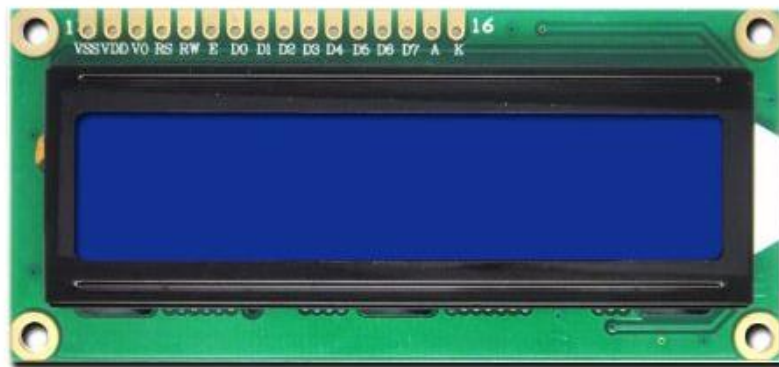
Srdcem zařízení je Arduino Nano. Toto Arduino bylo zvoleno z mnoha dalších variant pro svoje nejpříznivější vlastnosti. Je osazen mikrokontrolérem AT-Mega 328. Má malé rozměry (45x18mm), takže lze použít přímo do DPS. Pinová výbava činí 14 Digitálních pinů z toho 6 přizpůsobených pro PWM, 8 Analogových a další napájecí piny. Klidový odběr činí 19mA a piny dokáží sepnout 40 mA. Vstupní napětí se pohybuje mezi 7-12V. Paměť procesoru je 32KB z toho 2KB zabírá bootloader.



11 Arduino Nano

3.2.2 LCD Displej

Jako zobrazovač byl zvolen displej QAPASS 1602a. Jde o LCD displej s řadičem, který má mez zobrazení 2x16 znaků (16 bodů v řadě a 2 řádky). Displej je podsvícený modře s bílými znaky a přes přídavný modul může komunikovat přes rozhraní I²C. Rozhraní Arduino obsahuje knihovnu pro ovládání LCD displejů, tudíž je jednoduché ho k Arduinou připojit.



12 LCD displej

3.2.3 Stabilizátor napětí

Napájecí napětí Arduino je 7-12V. Protože palubní napětí většiny strojů se pohybuje mezi 12-15V, bylo zvoleno napájecí napětí 8V, které je z palubního napětí stabilizováno stabilizátorem 7808-STM. Jeho výstupní proud je 750mA a Díky vstupnímu napětí až 35V může být otáčkoměr připojen i do nákladních automobilů, ve kterých se palubní napětí pohybuje mezi 24-28V.

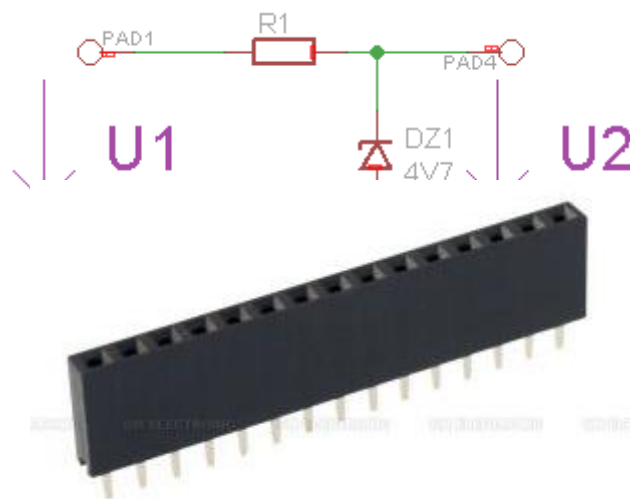


13 Stabilizátor napětí

3.2.4 Ochrana vstupních pinů

V případě použití u zážehových motorů mohou být impulsy snímány s přerušovače zapalovací cívky. Tam ve špičce vzniká napětí až kolem 150V. Proto byla zvolena ochrana Zenerovým stabilizátorem napětí. Jeho činnost spočívá na principu Zenerovy diody. Ta se používá v závěrném směru a udržuje na sobě tzv. Zenerovo napětí (v tomto případě 4,7V). Před Zenerovu diodu se řadí vhodný rezistor (v tomto zapojení 1k), který omezuje proud procházející diodou. Vznikne dělič napětí. Tato ochrana není vhodná pro měření vysokých frekvencí, stabilizátor by nemusel stíhat správně stabilizovat délku periody kvůli vysoké paralelní kapacitě Zenerovy diody.

3.2.5 Dutinkové lišty

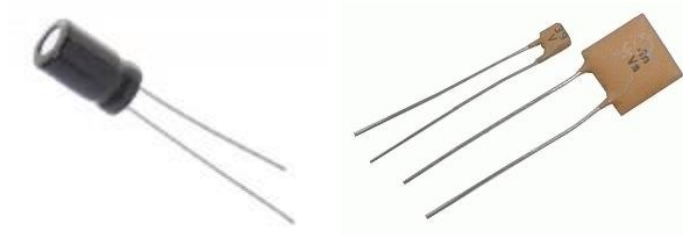


15 Dutinková lišta

Z důvodu potřeby vyjímání Arduina z DPS kvůli programování bylo nutné realizovat jednoduchý, ale spolehlivý spoj. Jelikož má Arduino vyvedené piny lámací lištou, padla volba na dutinkové lišty. Což je vlastně protikus lámací lišty (jumperu).

3.2.6 Kondenzátory

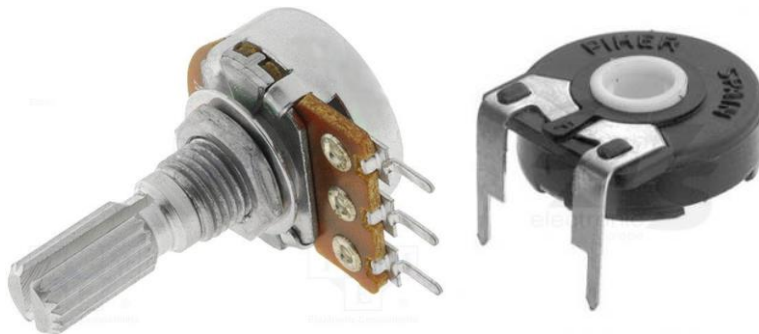
Další součástí jsou kondenzátory. Jde o součástku, která je schopna podle kapacity udržet elektrický náboj. Další vlastností kondenzátorů je, že pro střídavé napětí se chovají jako zkrat. Proto v tomto zapojení slouží k zachycení napěťových špiček v napájecím napětí. Jsou zde použity dva keramické a jeden elektrolytický.



16 Elektrolytický a keramický kondenzátor

3.2.7 Potenciometr a trimry

V zapojení je také užito dvou uhlíkových trimrů a jednoho potenciometru. Jde o součástky, které při otáčení osou mění poměr odporu na její dráze. Potenciometr byl zvolen pro regulaci jasu displeje, protože ho na rozdíl od trimru lze ovládat bez použití nástroje.



17 Potenciometr a trimr

3.2.8 Magnetický snímač

Jako snímač otáček byl zvolen skleněný magnetický snímač. Je to vlastně Jazyčkové relé, které není spínáno cívkou, ale magnetickým polem permanentního magnetu.



18 Magnetický snímač

3.2.9 Ochrana proti přepólování

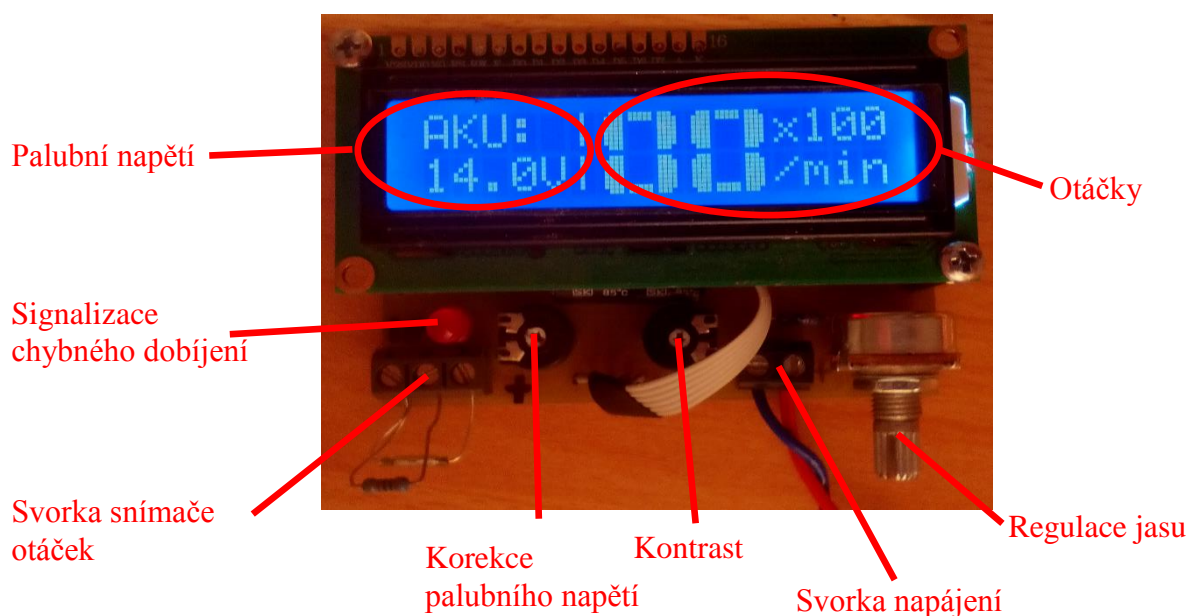
Jako ochrana proti možnému přepólování napájecího napětí a zničení řídicí elektroniky byla zvolena křemíková dioda v SMD provedení DC M1. V zapojení je využito propustnosti diody. Je zapojena sériově s obvodem na „+“ větvi v propustném směru. Pokud je zapojeno napájecí napětí se správnou polaritou, dioda je v propustném směru a prochází jí proud, na diodě vzniká úbytek 0,7V. Pokud se obrátí polarita, je dioda zapojena v závěrném směru a proud neprojde.



19 Dioda

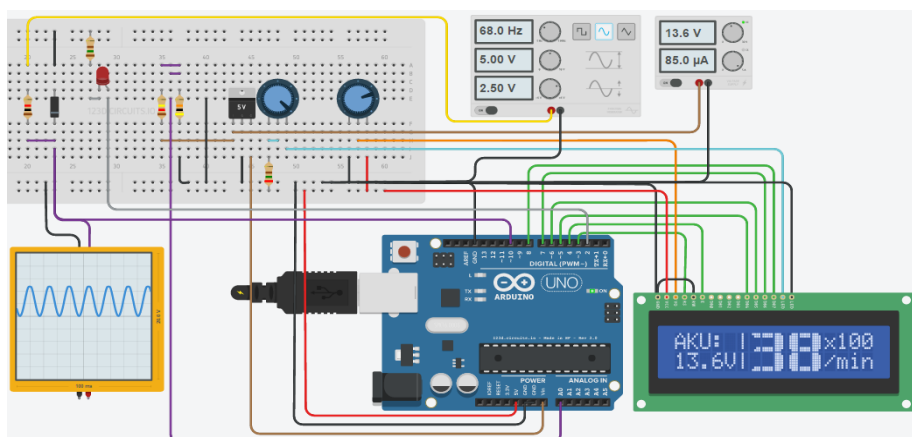
3.4 Popis činnosti

Zařízení je napájeno z palubní sítě vozidla do svorkovnice K2. Po připojení napájecího napětí prochází proud přes diodu D1, která chrání obvod proti prepólování. Napětí je stabilizováno stabilizátorem IO1 a vyfiltrováno kondenzátory C1; C2; C3. Vyfiltrované napětí je vedeno na napájecí piny Arduina a napěťový dělič složený z odporů R4; R5 a trimru P1. Ten má za úkol přizpůsobit napájecí napětí na maximální hodnotu 5V a vést ho do analogového pinu A0, na kterém procesor přečte počet bitů přímo úměrný hodnotě napětí a jednou za 100ms zobrazí na LCD display skutečnou hodnotu palubního napětí. V případě že palubní napětí klesne pod 13,8V nebo vzroste nad 14,4V, nastane na digitálním pinu 5 stav HIGH a přes rezistor R5 se rozsvítí červená LED dioda D2, která signalizuje špatné dobíjení. Snímač otáček je připojen do svorkovnice K1 kde se přes Zenerův stabilizátor složený z rezistoru R3 a Zenerovy diody DZ1. omezí impulsy na maximální hodnotu 4,7V. Dále jsou vedeny na digitální vstup 13, kde je měřena frekvence metodou měření délky periody. Ta se následně převede na otáčky a zobrazí na displeji stejným způsobem jako palubní napětí. Na výstupech 6 až 11 jsou připojeny vodiče pro přenos zobrazovacích dat na displej. Napájení displeje je realizováno z 5V napájecího výstupu Arduina, nastavení kontrastu je připojeno přes trimr P2 a jas je nastavován pomocí kombinace odporu R6 a potenciometru P3. Regulace jasu je založena na principu omezování proudu do podsvícení displeje Viz přílohy – Schéma zapojení.



3.5 Simulace a vývoj

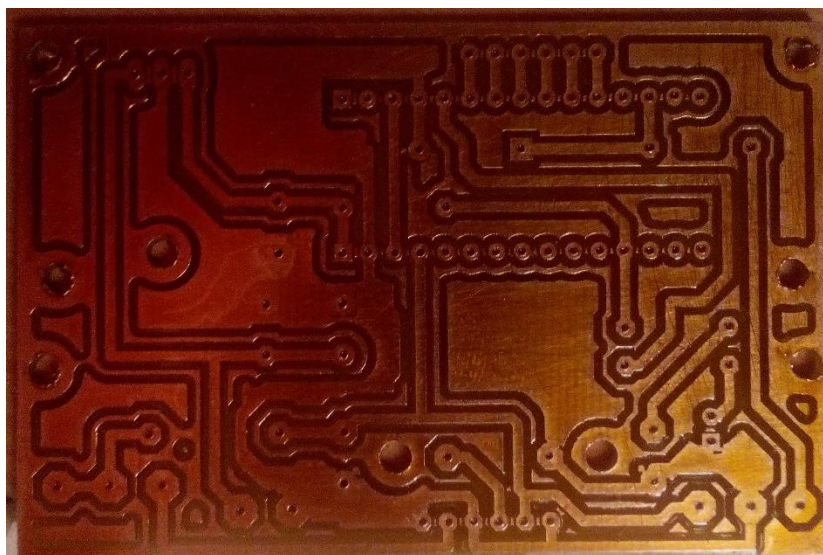
Pro vývoj zařízení bylo použito nepájivého kontaktního pole a simulačního programu Autodesk na webových stránkách <https://circuits.io>. Tento simulační program je vybaven několika deskami Arduino, širokým výběrem elektronických součástek, Arduino shieldů a vstupních i výstupních laboratorních přístrojů, jako jsou například Osciloskop, generátor tvarovaných průběhů a další. Základní stavba probíhá na virtuální nepájivé pole. Na něj se vkládají potřebné součástky a propojují vodiči různých barev dle potřeby. Propojení pinů je jako u reálného nepájivého pole. Výhodou simulátoru je pohodlí při vývoji zařízení a v případě nesprávného zapojení nedojde k fyzickému zničení součástky. Bohužel některá místa simulátoru nejsou úplně spolehlivá, proto bylo po od simulování rozhodnuto sestavit obvod ještě na reálné nepájivé pole a za pomoci reálných přístrojů doladit veškeré součástky. Po zdárném odzkoušení a změření funkcí byla zahájena výroba DPS.



22 Zapojení v simulátoru

3.6 Deska plošných spojů

Deska plošných spojů (DPS) je laminátová destička potažená slabou vrstvou mědi. Je to základní stavební prvek elektronických zařízení. Do měděné vrstvy se podle návrhu vygravírují, nebo vyleptají cesty, kterými jsou spojené jednotlivé součástky, a tím vznikne zařízení. Deska plošných spojů byla společně se schématem navržena v editoru schémat a tištěných spojů Eagle od firmy CadSoft. Byla navržena v co nejmenších možných rozměrech a se snadnou dostupností ke všem svorkám a ovládacím a nastavovacím prvkům. Zhotovení samotného DPS probíhalo za pomoci Tomáše Ryšavého na Gravírovací lince ve školních elektro dílnách u pana Ing Ladislava Kotrby. K ošetření DPS byla použita kalafuna rozpuštěná v technickém lihu. Tento roztok po zaschnutí vytvoří na povrchu malý film, který ochrání DPS před oxidováním a zlepší vlastnosti při pájení plošného spoje. Při návrhu DPS vznikly drobné chyby, které bylo možné jednoduše odstranit. První chyba zavinila, že jeden pól děliče měření palubního napětí byl připojen až za stabilizátorem, to způsobovalo chybnou funkci měření palubního napětí. Odstranění bylo možné provést lehkým přemostěním cest. Druhá chyba byla zapomenutá ochrana proti přepólování. Tu bylo možné díky konstrukci DPS vložit do „+“ cesty v provedení SMD. V příloze je DPS i s osazovacím plánem již opraveno.



23 Plošný spoj

3.7 Uvedení do provozu

Zařízení by mělo být do vozidla připojeno tak, aby při vypnutí klíčku neodebíralo žádný proud a zároveň bylo jištěno pojistkami palubní sítě (pojistky označené číslem 15). Po zapnutí klíčku zapalování se na svorkovnici K2 objeví napětí a rozsvítí se displej. Na něm se nejprve zobrazí „pevné“ údaje (Jednotky měřených veličin), a asi po 100ms prodlevě se objeví i „plovoucí“ údaje (otáčky a palubní napětí). Při překročení rozsahu otáčkoměru se zařízení vynuluje a nedochází k zobrazení chybného údaje. Trimrem P2 nastavíme kontrast tak aby bylo zřetelné zobrazení znaků a potenciometrem P3 nastavíme nám příjemný jas displeje. Nakonec pomocí voltmetru a otáčením trimru P1 na kalibrujeme zobrazované napětí a po připojení snímače otáček můžeme nastartováním motoru otestovat funkci přístroje.



24 Oživování

4. Zajímavé části zdrojového kódu

Zdrojový kód, nahraný v čipu Arduina řídí celé zařízení je sestaven z několika částí. Byl napsán v jazyce C ve vývojovém prostředí Arduino IDE. Principy zajímavých částí jsou popsány níže.

4.1 Měření otáček

Měření otáček je realizováno příkazem „pulseIn“. Příkaz obsahuje v závorce dva parametry: pulseIn(číslo pinu, sledovaná úroveň signálu); Podle zadaného parametru (HIGH/LOW) měří dobu trvání úrovně signálu v milisekundách. V dalším kroku se do proměnné „frekvence“ přiřadí skutečná hodnota frekvence v Hz. Ta se vypočítá podle známého vzorce a vynásobí 1000000. Poslední krok měření otáček převede přes konstantu frekvenci na otáčky.

4.2 Měření napětí

```
z = pulseIn(10, HIGH);
zz = pulseIn(10, LOW);
x = pulseIn(10, HIGH); //měření části HIGH
y = pulseIn(10, LOW); //měření části LOW
frekvence= (1000000/(float)(x+y));
//výpočet podle vzorce f=1/T, kde x+y je délka periody
otacky=frekvence*(0.6); //převod frekvence na otáčky
```

Měření palubního napětí zajišťuje příkaz „analogRead“ Tento příkaz obsahuje pouze jeden parametr a to je číslo analogového pinu a může fungovat pouze na analogových pinech. V prvním kroku se do proměnné „napeti“ přiřadí hodnota v bitech přečtená na analogovém vstupu A0. Po provrvení druhého kroku bude již proměnná obsahovat

```
napeti = analogRead(A0);
napeti = ((napeti*20)/1023);
```

skutečnou hodnotu napětí. Arduino obsahuje 8bitový A/D převodník, což umožní měřené napětí rozdělit na 1024 dílků. Výpočet se provede tak, že naměřená hodnota v bitech se vynásobí požadovaným rozsahem voltmetru a vydělí maximálním počtem dílků.

4.3 Zobrazení otáček

Zobrazování otáček na displeji ve velkém fontu se děje pomocí procedury „printOtacky“. V proceduře jsou deklarované proměnné „d“ a „u“ tam se zapisují číslice, které se mají zobrazit. Proměnná „m“ je kvůli překročení rozsahu do proměnné „number“ se zapíše hodnota otáček a v následující podmínce se rozdělí první dvojčíslí do proměnných „d“ (řád tisíců) a „u“ (řád stovek). Zobrazení na displeji je aktualizováno jednou za 100ms. Pokud je frekvence vyšší než 166Hz což odpovídá asi 9900 ot/min zobrazí se na 6. a 9. sloupci displeje proměnná „m“, která je rovna 0. To signalizuje překročení rozsahu. Pokud je frekvence nižší než 166Hz provádí se větve „else“ a ta zobrazí na 6. sloupci proměnnou „d“ a na 9. sloupci proměnnou „u“.

```
void printOtacky(int otd) {  
  
    int d, u, m, number;  
    m=0;  
    number = otd;  
    if (number > 9) {  
        d = (number - (number % 10)) / 10;  
        number = number % 10;  
    }  
    else {  
        d = 0;  
    }  
    u = number;  
    if (millis() > (100 + ukonceni))  
    {  
        if(frekvence>166){  
            printNumber(m, 6);  
            printNumber(m, 9);  
        }  
        else{  
            printNumber(d, 6);  
            printNumber(u, 9);  
        }  
    }  
}
```

4.4 Zobrazení napětí

Napětí je na displeji aktualizováno stejným způsobem jako otáčky motoru. Nezobrazuje se však velkými číslicemi ale standardním formátem pod nápisem „AKU“. První příkaz nastaví kurzor na 1. sloupec a 2. řádek displeje (souřadnice začínají v levém horním rohu displeje bodem [0,0]). Druhý a třetí krok je samotné měření napětí, které je popsáno výše. Čtvrtý řádek je příkaz pro zobrazení údaje na displeji. Příkaz obsahuje přetypovanou proměnnou „napeti“ ve které je obsažen údaj o aktuálním palubním napětí. Jednička za čárkou značí počet desetinných míst zobrazovaného údaje.

V posledním kroku je řešena signalizace nesprávného dobíjení. Pokud je proměnná „napeti“ větší než 13,8V a zároveň menší než 14,4V, jde o správnou hodnotu dobíjení a příkazem „digitalWrite se do pinu „dobijeni“ na kterém je připojena červená LED dioda přiřadí logická hodnota LOW (0). Pokud tato podmínka splněná není (napětí je menší než 13.8V nebo větší než 14.4V) vozidlo nedobíjí správně a na pin „dobijeni“ je přiřazena logická hodnota HIGH (1). Příkaz „digitalWrite“ může fungovat pouze na digitálních pinech a obsahuje dva parametry: digitalWrite(číslo/název pinu, hodnota LOW/HIGH).

```
LCD.setCursor(0,1);
napeti = analogRead(A0);
napeti = (napeti*20)/1023;
LCD.print(float(napeti),1);
  if((napeti>13.8)&&(napeti<14.4)){
    digitalWrite(dobijeni,LOW);
  }
  else{
    digitalWrite(dobijeni,HIGH);
  }
```

4.5 Zobrazení jednotek

Zobrazení jednotek, je méně zajímavá část kódu jsou to „pevné“ údaje, které jsou na displeji zobrazeny trvale. Jde o nastavování kurzoru pomocí příkazu „LCD.setCursor“ a vypisování znaků pomocí „LCD.print“. Na prvním řádku je nastaven kurzor na souřadnice [0,0] (1. sloupec, 1. řádek) a následným příkazem vypsán text: „AKU:“. Tento text označuje část displeje, na které bude zobrazeno palubní napětí. Následující dva symboly „|“ jsou vykreslovány na souřadnicích [5,0] a [5,1] Jsou tak zobrazeny dvě svislé čáry, které opticky oddělují část palubního napětí a otáček motoru. Poslední částí je zobrazení jednotek měřených veličin. Na souřadnicích [4,1] (5. sloupec, 2. řádek) je zobrazeno velké písmeno „V“, které značí jednotku palubního napětí (volty). Na konci prvního řádku (souřadnice [12,0]) je vypsán násobek zobrazovaných otáček a na druhém řádku (souřadnice [12,1]) je vypsán časový interval, ve kterém jsou otáčky motoru měřeny. Pokud tedy na displeji bude svítit velké číslo 36, budou otáčky motoru 3600 ot./min.

```
LCD.setCursor(0,0);
LCD.print("AKU:");
LCD.setCursor(5,0);
LCD.print("|");
LCD.setCursor(5,1);
LCD.print("|");
LCD.setCursor(4,1);
LCD.print("V");
LCD.setCursor(12,0);
LCD.print("x100");
LCD.setCursor(12,1);
LCD.print("/min");
```

4.6 Velký font číslic

Kód pro zobrazení velkého fontu číslic byl převzat a upraven z webových stránek https://exploreembedded.com/wiki/Distance_Meter_with_Big_Fonts Jde o použití jednoho znaku displeje jako jednoho segmentu číslice. Tak vznikne velké číslice, která se rozprostře přes tři sloupce displeje.

4.6.1 Části číslic

Jednotlivé části číslic jsou sestaveny ze segmentů, které jsou vykresleny do políčka displeje. Políčko displeje má rozměr 5x8 pixelů. Pro ovládání segmentu je použito 8bitové pole, ve kterém je vyplněn tvar segmentu. 1 = pixel rozsvícen, 0= pixel zhasnut. Příklad kódu je uveden z pole číslo 1. jde o pravou část čísla 3 nebo 8.

```
byte bar1[8] = //segmenty číslic
{
    B11100, // 1=pixel svítí
    B11110, // 0=pixel nesvítí
    B11110,
    B11110,
    B11110,
    B11110,
    B11110,
    B11100
};
```

4.6.2 Stavba číslic

Jednotlivé číslice jsou stavěny v procedurách z předem definovaných polí. Příklad ukazuje stavbu číslice 0. Kód nastaví kurzor na první řádek displeje, na kterém zobrazí pole 2;8;1. tím vznikne horní část číslice v prvním řádku. Dále nastaví kurzor do druhého řádku a tam zobrazí pole 2;6;1. tím vznikne spodní část číslice a na displeji můžeme sledovat velkou číslici. V tomto případě nulu.

```
void custom0(int col) // sestavení segmentů
{
    LCD.setCursor(col, 0); //první řádek
    LCD.write(2); // zobrazí pole2
    LCD.write(8);
    LCD.write(1);
    LCD.setCursor(col, 1); //druhý řádek
    LCD.write(2);
    LCD.write(6);
    LCD.write(1);
}
```

4.6.3 Přřazení číslic k hodnotě

Procedura „printNumber“ přiřazuje podle hodnoty otáček číslici, která se má zobrazit na displeji. Výsledek této části kódu je využíván v proceduře „printOtacky“, kde proměnné „m“, „d“ a „u“ jsou hodnoty proměnné „value“ procedury „printNumber“. Pokud jedna z proměnných bude mít hodnotu 6, bude výsledkem „custom6(col)“. To odpovídá číslici 6, která se v proceduře „printNumber“ vytiskne na displeji ve velkém fontu.

```
void printNumber(int value, int col) {
    if (value == 0) {
        custom0(col);
    } if (value == 1) {
        custom1(col);
    } if (value == 2) {
        custom2(col);
    } if (value == 3) {
        custom3(col);
    } if (value == 4) {
        custom4(col);
    } if (value == 5) {
        custom5(col);
    } if (value == 6) {
        custom6(col);
    } if (value == 7) {
        custom7(col);
    } if (value == 8) {
        custom8(col);
    } if (value == 9) {
        custom9(col);
    }
}
```


5. Další možná řešení

5.1 Snímání indukčním čidlem

Otáčky lze mimo jiné měřit indukčním snímačem. Jde o podobný princip jako magnetickým snímačem, ale liší se nutností napájení a podmínkami sepnutí. Indukční snímač pracuje na principu magnetického indukčního pole. Pokud má v blízkosti jakoukoliv feromagnetickou látku, změní se stav na výstupu snímače. Většina rotačních částí v motorovém prostoru je vyrobena z kovu, proto je složité upravit je tak aby snímač mohl číst a zároveň nebyly poškozeny mechanické vlastnosti těchto částí. Kvůli výstupnímu napětí zařízení (5V) není snadné takové čidlo dostat. Proto by tato varianta byla složitější a nákladnější

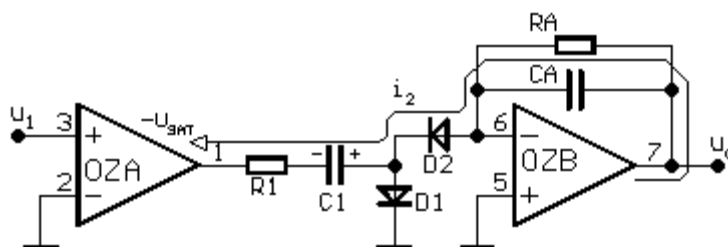


25 Indukční snímač

5.2 Realizace převodníkem f/U

Tato možnost byla jedna z prvních, kterými měl být otáčkoměr realizován. Nakonec však od tohoto řešení upadlo, kvůli velké DPS a složitosti při změnách konstant měření. Převodník f/U je vlastně zdroj napětí řízení frekvencí. Vyrábí se jako integrovaný obvod nebo ho lze sestavit z operačních zesilovačů.

První OZ funguje jako tvarovací obvod, který tvaruje vstupní signál na pravoúhlý. Druhý OZ je řídicí obvod. Ten podle frekvence na vstupu udržuje za pomoci Zpětné vazby z kondenzátoru a odporu napětí úměrné frekvenci.



26 Převodník f/U

6. Závěr

V dnešní době není potřeba se obávat používání programovatelných součástek. Jejich vývojová prostředí jsou dnes na takové úrovni, že je lze jednoduše ovládat a odladit chyby

ještě před samotným nahráním. Mnohdy je s nimi jednodušší práce než s obyčejnými integrovanými obvody.

Při měření frekvencí je třeba brát v potaz rostoucí odchylku, která roste společně s frekvencí. Proto je důležité zvolit nejvhodnější metodu měření. U výrobku nebylo nutné brát ohled na rostoucí odchylku, protože jeho akční rozsah je 0-165Hz. Vyšší frekvence jsou stále měřeny, ale programově je ošetřeno, aby nedošlo k jejich zobrazení na displeji. Metoda měření délky periody byla vybrána kvůli jednoduché programové realizaci příkazem Pulse, které spolehlivě funguje do 1kHz.

7. Zdroje informací

Zápisky ze školní docházky 2013-2017

<https://cs.wikipedia.org/wiki/Frekvence>

https://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=66944

http://www.outech-havirov.cz/Skola/files/knihovna_eltech/me/me_vyb_kap_all_2r.pdf

<https://www.google.cz/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0ahUKEwiR8OrI->

<http://vbRAhXC1hoKHRpnD80QFggbMAA&url=http%3A%2F%2Fwww.spse-fa.mzf.cz%2Fhome%2FESO%2520->

%2520Maturitni%2520otazky%2F30_mereni_kmitoctu_a_faze.odt&usg=AFQjCNGcynqlv-4pJFXCv_gdi5F1rq34SQ&sig2=8tiVhUfmfVbduT5CD3agag

<http://slideplayer.cz/slide/3106294/>

http://www.fm.tul.cz/esf0247/download_file.php/MT1_13_mereni_kmitoctu.pdf?id=940

<http://stepa.wz.cz/hobbies/GF/sinus.jpg>

<http://stepa.wz.cz/hobbies/BM556/komplet.jpg>

http://www.grandic.cz/photo/products/p36/p19836_1.jpg

<https://arduino.cz/seznameni-s-arduinem/>

<https://www.arduino.cc/en/Main/arduinoBoardNano>

<http://www.theengineerstore.in/Display-Modules/16x2-Character-LCD-Display---Blue-colour-id-2508334.html>

https://www.gme.cz/data/product/300_300/832-052.1.jpg

http://www.elektro-hofman.cz/obchody/elektro-hofman.shop5.cz/prilohy/8/876072_0.jpg

<http://www.prag->

info.cz/shop/resize_detailbig.php?image=image_0102000a/303870_2_303869_2_303868_2_303867_2_303866_2_303865_2_303864_2_303863_2_303862_2_303861_2_303860_2_303859_2_303858_2_303857_2_303856_2_303855_2_303854_2_139028923302040039.jpg

http://www.ame.cz/Potenciometr-mono-otocny-500k-B-c-56-linearni-drazkovana-hridel-6-x-13-5mm-R16110N-a32568810_10699.aspx

<http://www.prag->

info.cz/shop/resize_gzbozi.php?image=image_0102000a/310241_310240_1396509318pt15_nv.jpg

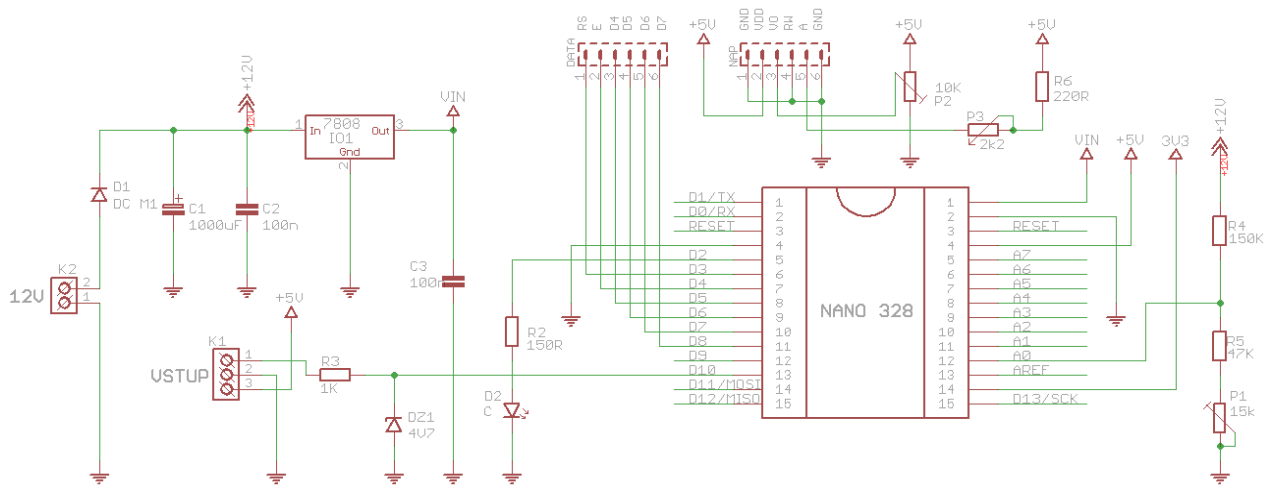
<https://www.gme.cz/jazyckovy-magneticky-senzor-ka-80>

https://explorembedded.com/wiki/Distance_Meter_with_Big_Fonts

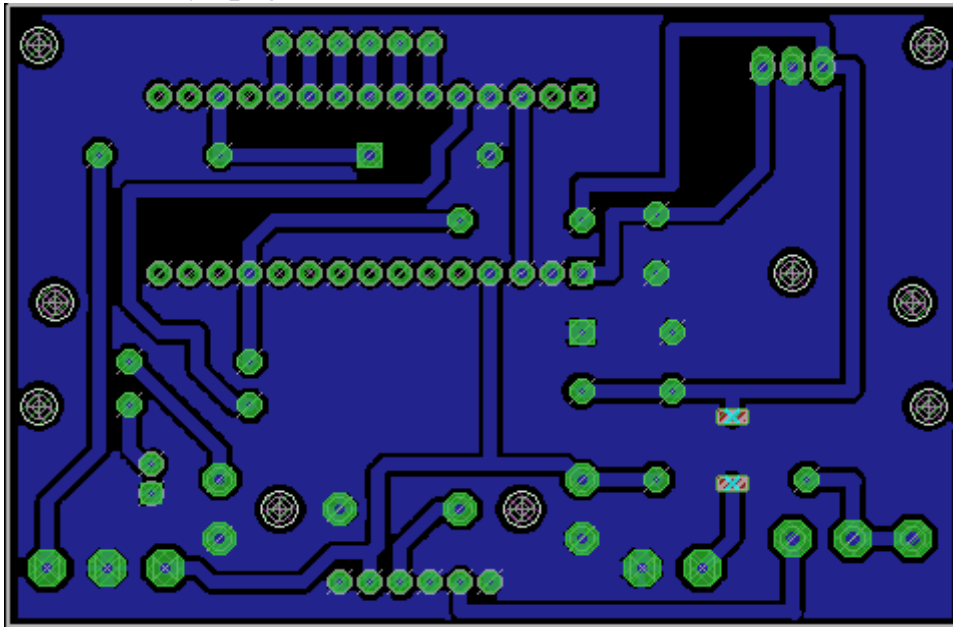
<http://www.cncshop.cz/ips-8-indukcni-snimac-m8x1-kabel-1-5m>

http://home.zcu.cz/~ronesova/index.php?menuitem=aes_mfu#Schema1

7.1 Schéma zapojení



7.2 Plošný spoj



7.3 Osazovací plán

